Assingment #4

JDBC & OLE DB/ADO.NET

KAIST

Myoung Ho Kim

Contents

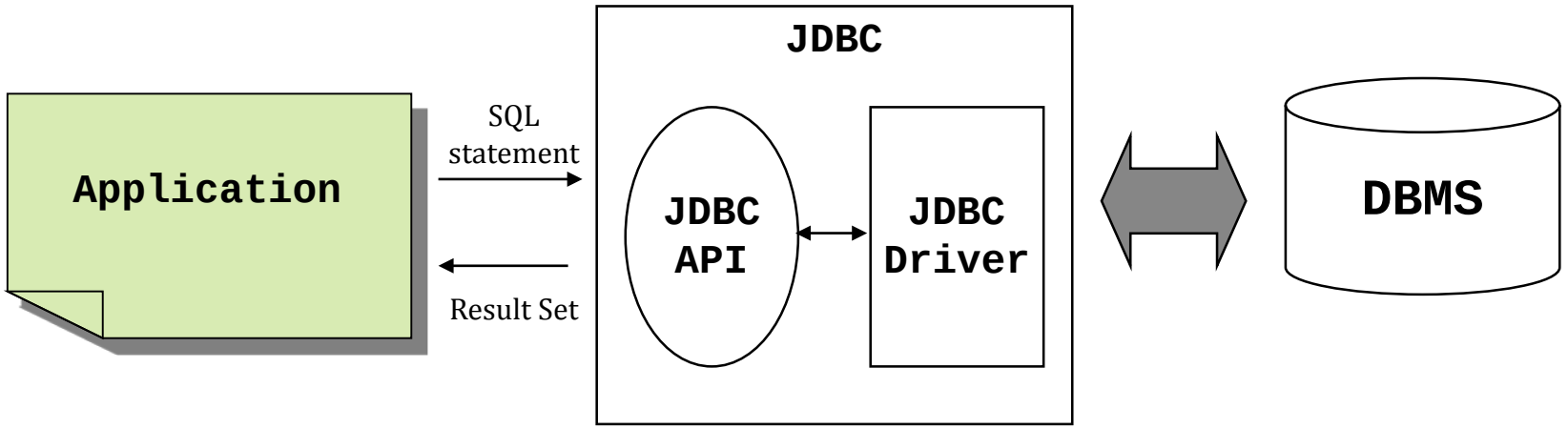
- Introduction to JDBC
  - Example
  - Main classes & methods
  - JDBC driver installation
- Introduction to OLE DB/ADO.NET
  - Example
  - Main classes & methods
  - .NET framework SDK and Oracle Client program installation
- HW Assignment
- Directions for HW
- References

JDBC

1. Introduction to JDBC
2. Example
3. Main classes & method
4. JDBC driver installation

Introduction to JDBC

- What is JDBC?
  - "Java Database Connectivity"
  - Connector to access DB, when developing applications in Java™ Platform



Example of JDBC code

```
import java.sql.*;

class Test  {
   public static void main(String[] args)  {
      Connection con = null;
      Statement stmt = null;

      try {
         Class.forName("oracle.jdbc.driver.OracleDriver");
         con = DriverManager.getConnection( "jdbc:oracle:thin:@dbclick.kaist.ac.kr:1521:orcl", "user", "passwd");
         stmt = con.createStatement();
         ResultSet rs = stmt.executeQuery("select name from product");

         while (rs.next()) {
            String product = rs.getString(1);
            System.out.println(product);
         }
      } catch (Exception e) {
         e.printStackTrace();
      } finally {
         try {
            if (stmt != null) stmt.close();
            if (con != null) con.close();
         } catch (Exception e) { }
      }
   }
}
```

You can download *example.java* from the course homepage

Main classes & method

- Loading JDBC driver
  - Using Class.forName()

```
Class.forName(“oracle.jdbc.driver.OracleDriver”);
```

- Connecting to DB
  - Using DriverManager.getConnection()

```
Connection con =
   DriverManager.getConnection("jdbc:oracle:thin:
     @dbclick.kaist.ac.kr:1521:orcl", "user", "passwd");
```

Main classes & method (cont'd)

- Executing queries
  - Using Statement class

```
Statement stmt = con.createStatement();
ResultSet rs = stmt.executeQuery("SELECT name FROM product");
```

  - Using PreparedStatement class

```
PreparedStatement pstmt =
   con.prepareStatement("INSERT INTO product values(?, ?)");
pstmt.setString(1, "mp3");
pstmt.setInt(2, 150);
pstmt.executeUpdate();
```

※ Use executeUpdate() for insert, update, and delete

Main classes & method (cont'd)

- Cursor operations
  - Use methods of ResultSet class
    - » Ex) next(), getString(), etc.

```
ResultSet rs = stmt.executeQuery(“SELECT name FROM product");
while (rs.next()) {
   String product = rs.getString(1);
   System.out.println(product);
}
```

Main classes & method (cont'd)

- Using 'finally'
  - Before finishing code, connection should be closed

```
try {
          …
    con = DriverManager.getConnection( … );
    stmt = con.createStatement();
          …
} catch (Exception e) {
    e.printStackTrace();
} finally {
   try {
          if (stmt != null) stmt.close();
          if (con != null) con.close();
   } catch (Exception e) {}
}
```

Main classes & method (cont'd)

- Executing Query within a Transaction

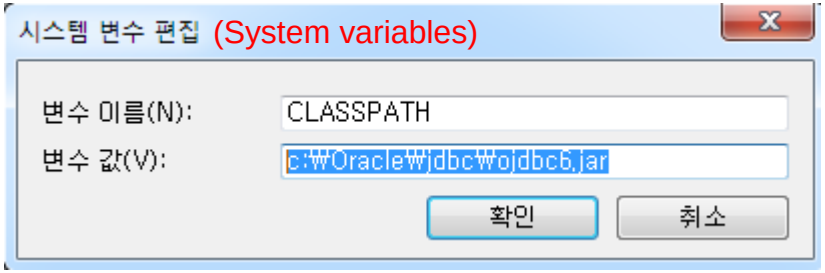
```
try {
          …
   con = DriverManager.getConnection( … );
   con.setAutoCommit(false);

   stmt = con.createStatement();
   stmt.executeQuery( … );
   stmt.executeQuery( … );
   …
   conn.commit();
} catch (SQLException e) {
   conn.rollback();
   …
}
```

JDBC driver installation

- JAVA SE 7.0 or 8.0 must be installed
  - See references
- Download (ojdbc6.jar)
  - http://www.oracle.com/technetwork/database/enterprise-edition/jdbc-112010-090769.html
  - or from the course homepage (KLMS)

Compile java using DOS command

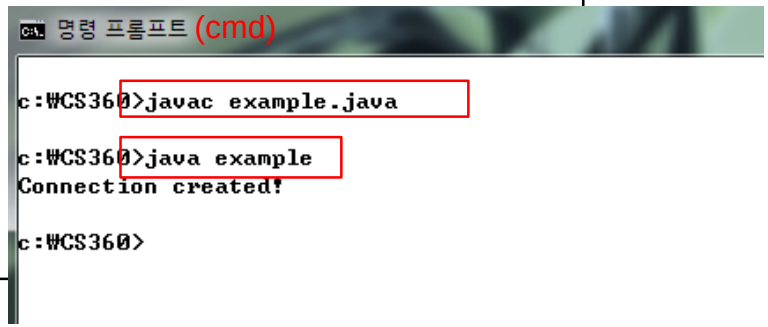
- Environment variable setting
  - » If you use the “Eclipse”, you don’t have to this setting
  - – Add(or create) to the CLASSPATH environment variable the driver installation path
    - » Ex) If the path is *c:\Oracle\jdbc\ojdbc6.jar*



Compile java using DOS command (cont'd)

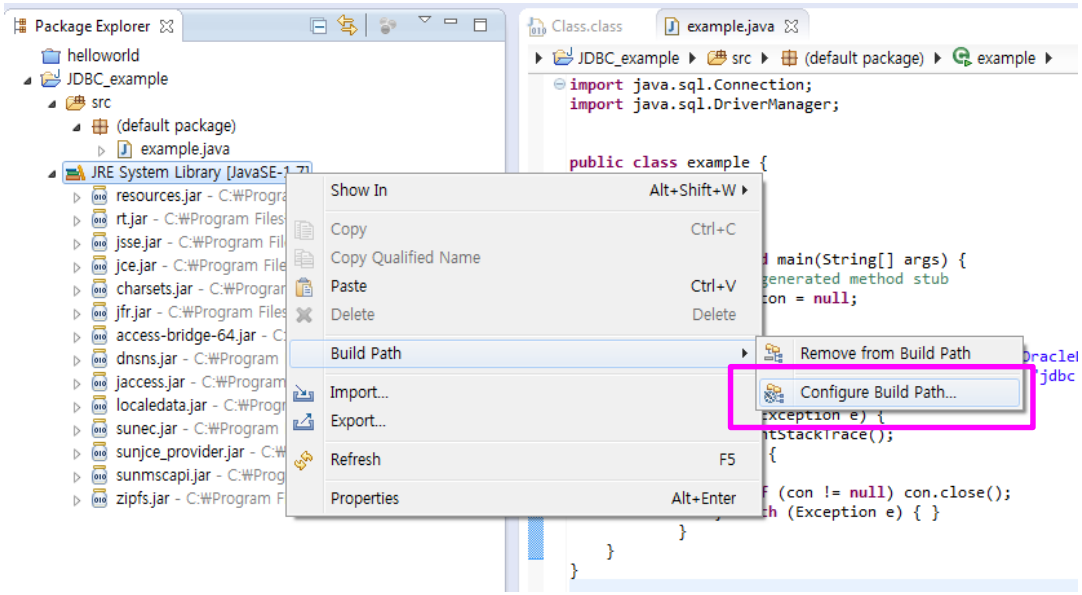
- Example file execution in the DOS command(cmd) window
  - Compiling & running

```
import java.sql.*;

class Test  {
   public static void main(String[] args)  {
      Connection con = null;

      try {
         Class.forName("oracle.jdbc.driver.OracleDriver");
         con = DriverManager.getConnection( "jdbc:oracle:thin:@dbclick.kaist.ac.kr:1521:orcl", "user", "passwd");
         System.out.println("Connection created");
      } catch (Exception e) {
         e.printStackTrace();
      } finally {
         try {
            if (con != null) con.close();
         } catch (Exception e) { }
      }
   }
}
```



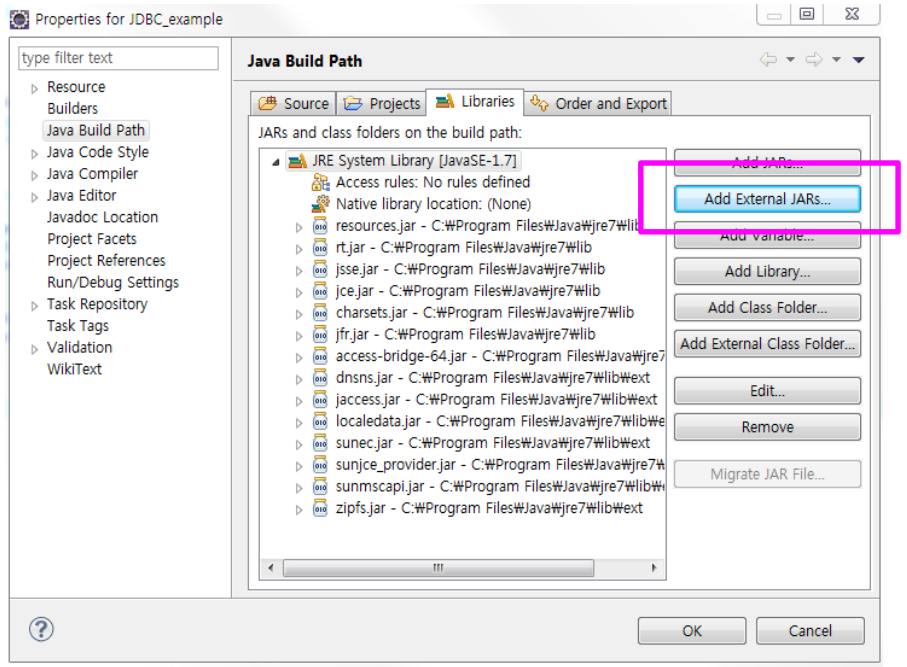
Compile java using Eclipse IDE

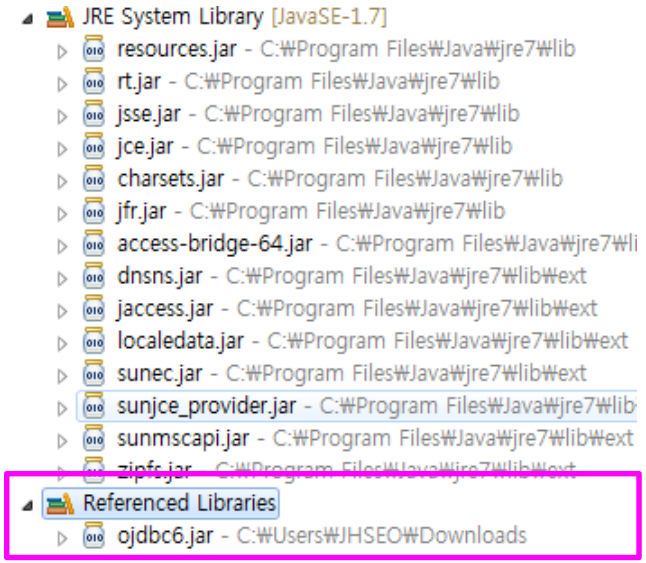
- Eclipse setting
  1. Add ojdbc6.jar to project build path
     - » Right click on JRE System Library → Build Path → Configure Build Path



Compile java using Eclipse IDE
(cont'd)

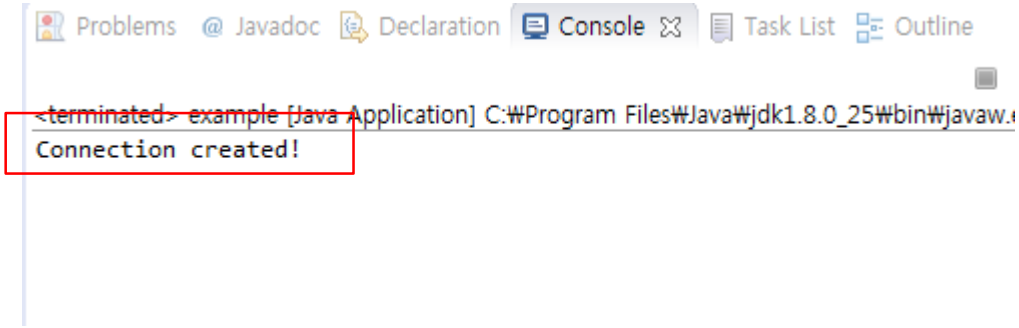
2. Add External IDE → select ojdbc6.jar





Compile java using Eclipse IDE
(cont'd)

- Example file execution in the Eclipse
  - Press "F5" key or click  to compile
    - » Then you can see



OLE DB/ADO.NET

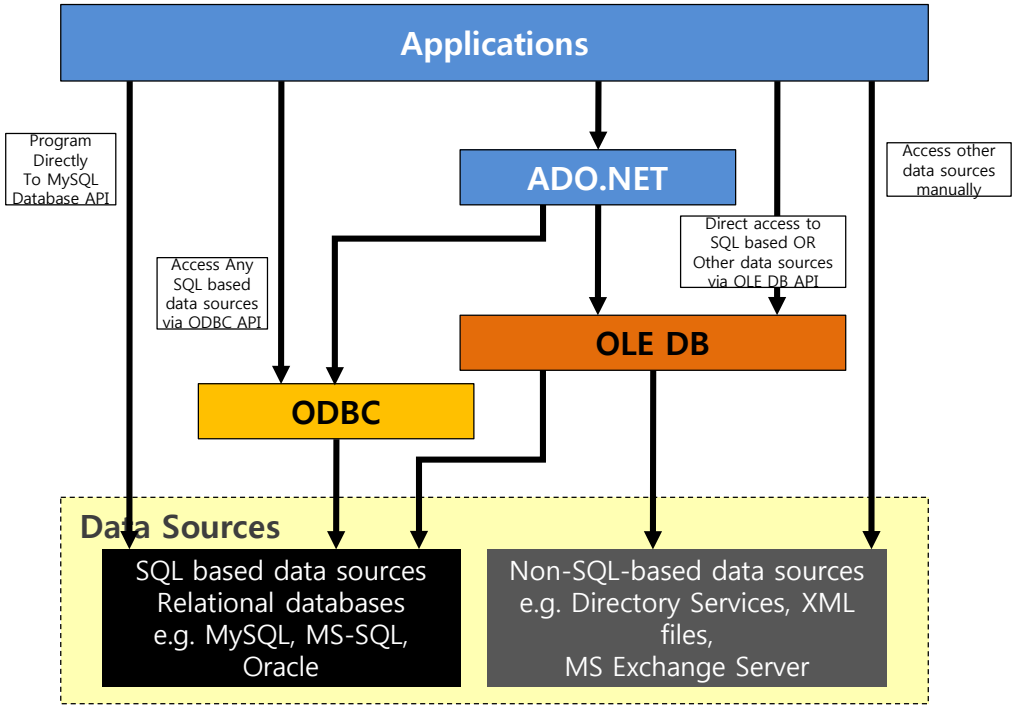
1. Introduction to OLE DB/ADO.NET
2. Example
3. Main classes & methods
4. .NET framework SDK and Oracle Client installation

Introduction to OLE DB/ADO.NET

- What is OLE DB?
  - An API for accessing different types of data stored in a uniform manner
  - A higher-level replacement for ODBC, extending features to support a wider variety of non-relational databases, such as text file, object databases and spreadsheet
  - Main objects
    - » CDataSource: connect to data sources
    - » CSession: manage interaction with data source
    - » CCommand: execute text commands like SQL statements
    - » CNoRowset: show data in tabular format

Intro to OLE DB/ADO.NET (cont'd)

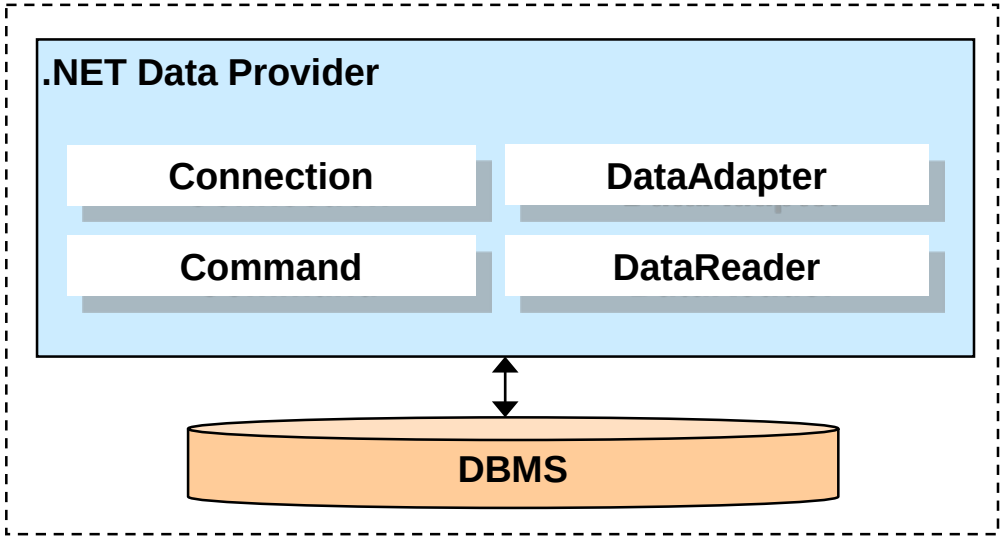
- What is ADO.NET?
  - It provides consistent access to data source, such as SQL Server, XML, Oracle, through OLEDB and ODBC.



To develop applications in the web environment!

Intro to OLE DB/ADO.NET (cont'd)

- .NET Data Provider
  - A set of objects required for interacting with Data Sources in ADO.NET



Object	Description
Connection	Establishes a connection to a specific data source
Command	Executes a command against a data source
DataReader	Reads data from a data source
DataAdapter	Resolves updates with the data source

Intro to OLE DB/ADO.NET (cont'd)

- .NET Data Provider (cont'd)
  - Types
    - » OLE DB .NET Data Provider → we use this
    - » ODBC .NET Data Provider
    - » SQL Server .NET Data Provider
  - Objects contained in OLE DB .NET Data Provider
    - » OleDbConnection (c.f. Connection of JDBC)
    - » OleDbCommand (c.f. Statement of JDBC)
    - » OleDbDataReader (c.f. ResultSet of JDBC)
    - » OleDbDataAdapter

Example of OLE DB/ADO.NET

Language: C#

```
using System;
using System.Text;
using System.Data.OleDb;

namespace OleDB_Test
{
   class Program
   {
      static void Main(string[] args)
      {
         OleDbConnection cn = null;
         try {
            cn = new OleDbConnection("Provider=OraOLEDB.Oracle;Data Source=CS360; User Id=username; Password=yourpasswd");

            cn.Open();

         }catch (Exception ex){
            Console.WriteLine("Error: Fail to connect to database");
         }finally{
            cn.Close();
         }
      }
   }
}
```

You can download *example.cs* from the course homepage

Main classes & method

- Connecting to DB
  - Using OleDbConnection

```
OleDbConnection cn =
        new OleDbConnection(“Provider=OraOLEDB.Oracle;
        Data Source=CS360;User ID=user;Password=passwd”);
cn.Open();
```

» Provider: name of .NET Data Provider

- : Oracle Provider OLE DB for Oracle

» Data Source: data source name

» User ID: user account

» Password : user password

Main classes & method (cont'd)

- Executing queries
  - For two or more resulting records
    - » Use ExecuteReader()

```
string strSQL = "select * from product";
OleDbCommand cmd = new OleDbCommand(strSQL, cn);
OleDbDataReader dr = cmd.ExecuteReader();
```

  - For one resulting record
    - » Use ExecuteScalar()

```
string strSQL = "select max(price) from product";
OleDbCommand cmd = new OleDbCommand(strSQL, cn);
decimal maxPrice = (decimal)cmd.ExecuteScalar();
```

※ Since the return type of ExecuteScalar() is Object, we need the casting operator '(decimal)'

Main classes & method (cont'd)

- Executing queries (cont'd)
  - For update, insert, delete
    - » use ExecuteNonQuery()

```
string strSQL = "update product set price = 100";
OleDbCommand cmd = new OleDbCommand(strSQL, cn);
int rnum = cmd.ExecuteNonQuery();
```

※ The returned value of ExecuteNonQuery() is the number of affected records

Main classes & method (cont'd)

- Parameterized query
  - Using Parameters

```
string name = “lcd”;
int price = 2500;
string strSQL = "insert into product values(?, ?)";
OleDbCommand cmd = new OleDbCommand(strSQL, cn);
// set type of parameter
cmd.Parameters.Add("name", OleDbType.VarChar, 10);
cmd.Parameters.Add("price", OleDbType.Integer);
// set value of each parameter
cmd.Parameters[0].Value = name;
cmd.Parameters[1].Value = price;
int rnum = cmd.ExecuteNonQuery();
```

  - Using String concatenation

```
string strSQL = "insert into product values('"+
          name+"',"+price+")";
OleDbCommand cmd = new OleDbCommand(strSQL, cn);
int rnum = cmd.ExecuteNonQuery();
```

Main classes & method (cont'd)

- Cursor
  - use OleDbDataReader.Read()

```
string strSQL = "select * from product";
OleDbCommand cmd = new OleDbCommand(strSQL, cn);
OleDbDataReader dr = cmd.ExecuteReader();
while(dr.Read()) {
① ----------> //System.Console.WriteLine(dr["name"]+"\t"+dr["price"]);
② ----------> //System.Console.WriteLine(dr[0]+"\t"+dr[1]);
③ ----------> System.Console.WriteLine(dr.GetString(0)+"\t"+dr.GetDouble(1));
}
```

※ ①, ②, and ③ lead to the same result

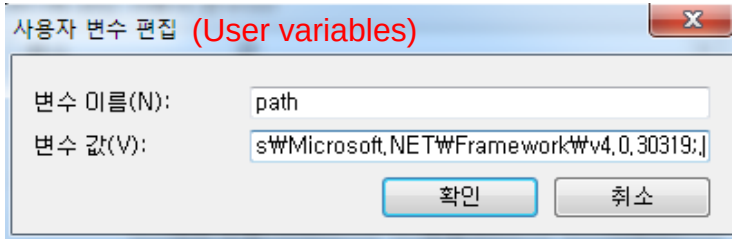
Main classes & method (cont'd)

- Executing Query within a Transaction

```
OleDBConnection cn;
OleDBTransaction txn;

try {
   cn = new OleDbConnection( … );
   cn.Open();

   txn = cn.BeginTransaction();

   cmd = new OleDBCommand("INSERT INTO …", cn, txn);
   cmd.ExecuteNonQuery();
   cmd = new OleDBCommand("INSERT INTO …", cn, txn);
   cmd.ExecuteNonQuery();
   …
   txn.Commit();
} catch (Exception e) {
   txn.Rollback();
}
```

.NET framework installation

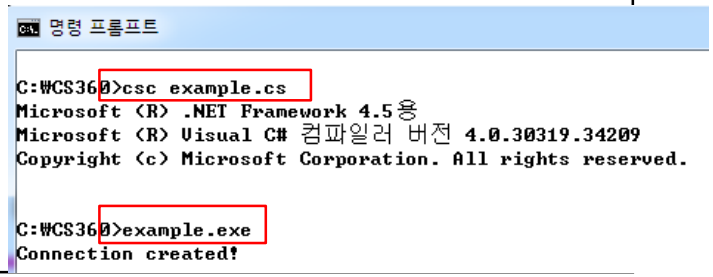
- Oracle Client program must be installed
  - If you didn't install the program, see assignment 1
- .NET framework SDK
  - Download (.NET Framework 4.5)
    - » http://www.microsoft.com/en-us/download/details.aspx?id=30653
  - Environment variable setting
    - » Add to the Path environment variable the installation path of .NET framework SDK
      - Ex) add *C:\WINDOWS\Microsoft.NET\Framework\v4.0.30319*



Compile C# using DOS command

- Example file execution in the DOS command(cmd) window
  - Compiling & running

```
using System;
using System.Text;
using System.Data.OleDb;

namespace OleDB_Test{
   class Program{
      static void Main(string[] args){
         OleDbConnection cn = null;

         try{
            cn = new OleDbConnection("Provider=OraOLEDB.Oracle; Data Source=CS360; User Id=s20150000; Password=TIGER");
            cn.Open();
            Console.WriteLine("Connection created!");
         } catch (Exception ex){
            Console.WriteLine(ex.Message);
            Console.WriteLine("Error: Fail to connect to database");
         } finally {
            cn.Close();
         }
      }
   }
}
```



Homework #4

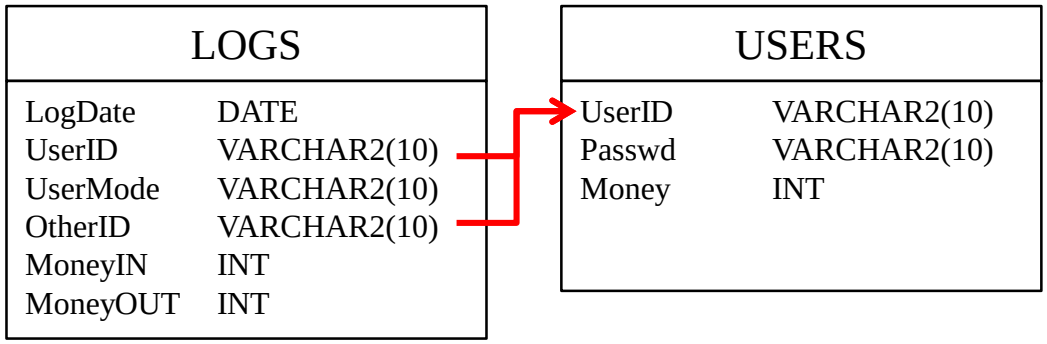
1. Overview
2. Homework Assignment
3. Directions
4. References

Overview

- *Simple Bank* implementation
  - manages “user accounts”
  - supports “deposit/withdraw/send money” commands
  - shows “user’s log”
  - transaction control
  - uses two tables *LOGS* and *USERS* in DB
    - » LOGS: contains user’s log
    - » USERS: stores user account info

Table creation

1. Download *HW4db.sql* from the course homepage and copy it to (directory that Oracle Client is installed)\BIN
2. Use the *SQLPlus* and perform the command *@HW4db.sql* or *start HW4db.sql*
   - » It contains only structures



Homework #4

- Initial screen
  - a user can login to Bank, or
  - create a new account by typing ‘new’

```
CS360   Simple Bank
please type 'new' to create  a new account.
user name:
```

Homework #4 (cont'd)

- HW4-1) Managing user accounts
  - Create a new account

```
CS360   Simple Bank
please type 'new' to create  a new account.
user name: new
ID: aaa
same ID exists
ID: bbb
Password: bbb
new account bbb is created.
CS360   Simple Bank
please type 'new' to create  a new account.
user name:
```

check if there exists the same ID

Homework #4 (cont'd)

- HW4-1) Managing user accounts (cont'd)
  - Login to BBS

```
CS360   Simple Bank
please type 'new' to create  a new account.
user name: aaa
password: b
invalid username/password
CS360   Simple Bank
please type 'new' to create  a new account.
user name: aa
password: asdf
invalid username/password
CS360   Simple Bank
please type 'new' to create  a new account.
user name: aaa
password: a
invalid username/password
unable to log-in to Simple Bank after 3 attempts
```

If a user type a wrong ID or password three times, login is denied and the program exits.

Homework #4 (cont'd)

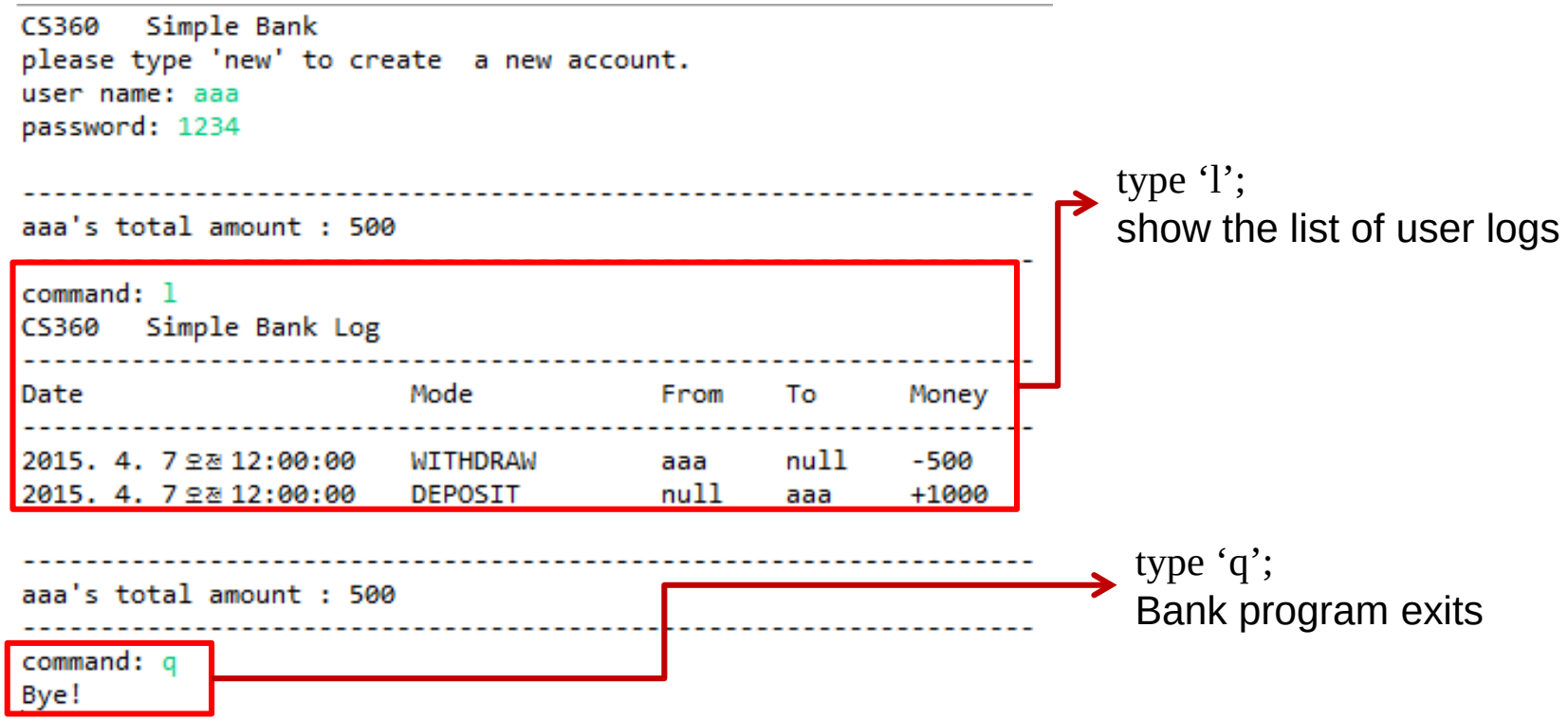
- HW4-1) Managing user accounts (cont'd)
  - If a login successes, show user's money

```
CS360   Simple Bank
please type 'new' to create  a new account.
user name: aaa
password: 1234

------------------------------------------------------------
aaa's total amount : 0
------------------------------------------------------------
command:
```

You set the default money = 0

Homework #4 (cont'd)

- HW4-2) Command: log(l)/quit(q)



Homework #4 (cont'd)

- HW4-3) Command: deposit(d) & transaction control

```
CS360   Simple Bank
please type 'new' to create  a new account.
user name: aaa
password: 1234

-------------------------------------------------------------------
aaa's total amount : 1100
-------------------------------------------------------------------
command: d
Deposit Moeny : 200

-------------------------------------------------------------------
aaa's total amount : 1300
-------------------------------------------------------------------
command: l
CS360   Simple Bank Log
-------------------------------------------------------------------
Date                    Mode            From    To      Money
-------------------------------------------------------------------
2015. 4. 7 오후 12:00:00    DEPOSIT         null    aaa     +1000
2015. 4. 7 오후 12:00:00    WITHDRAW        aaa     null    -500
2015. 4. 7 오후 12:00:00    DEPOSIT         null    aaa     +600
2015. 4. 7 오후 12:00:00    DEPOSIT         null    aaa     +200
```

type 'd'; input money;
show the result &
record this result in LOGS

You should handle an inconsistent money input
Ex) 0 or -1000 or …

Homework #4 (cont'd)

- HW4-4) Command: withdraw(w) & transaction control

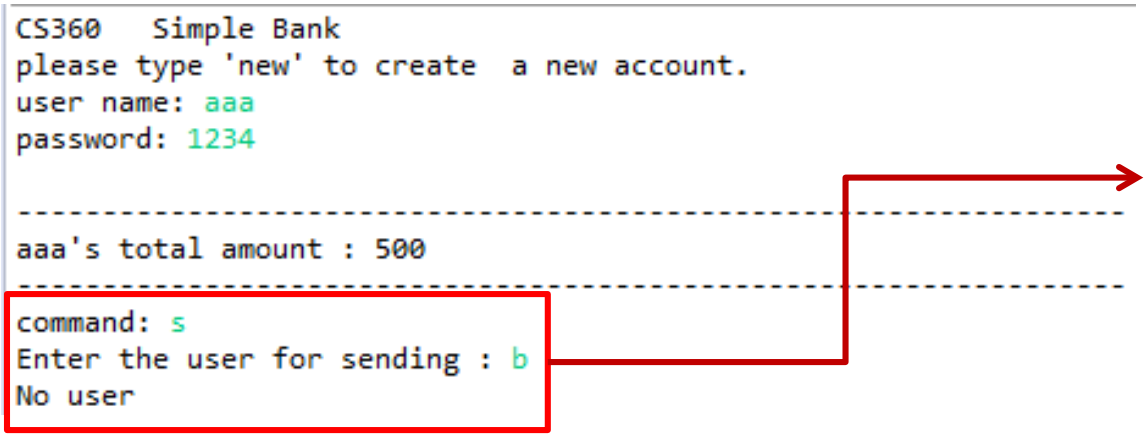
```
CS360   Simple Bank
please type 'new' to create  a new account.
user name: aaa
password: 1234

------------------------------------------------------------------
aaa's total amount : 1300
------------------------------------------------------------------
command: w
Withdraw Moeny : 800

------------------------------------------------------------------
aaa's total amount : 500
------------------------------------------------------------------
command: l
CS360   Simple Bank Log
------------------------------------------------------------------
Date                    Mode            From    To      Money
------------------------------------------------------------------
2015. 4. 7 오전 12:00:00    WITHDRAW        aaa     null    -800
2015. 4. 7 오전 12:00:00    DEPOSIT         null    aaa     +1000
2015. 4. 7 오전 12:00:00    WITHDRAW        aaa     null    -500
2015. 4. 7 오전 12:00:00    DEPOSIT         null    aaa     +600
2015. 4. 7 오전 12:00:00    DEPOSIT         null    aaa     +200
```

type 'w'; input money;
show the result &
record this result in LOGS

You should handle an inconsistent money input

Homework #4 (cont'd)

- HW4-5) Command: send(s) & transaction control



Homework #4 (cont'd)

- HW4-5) Command: send(s) & transaction control (cont'd)

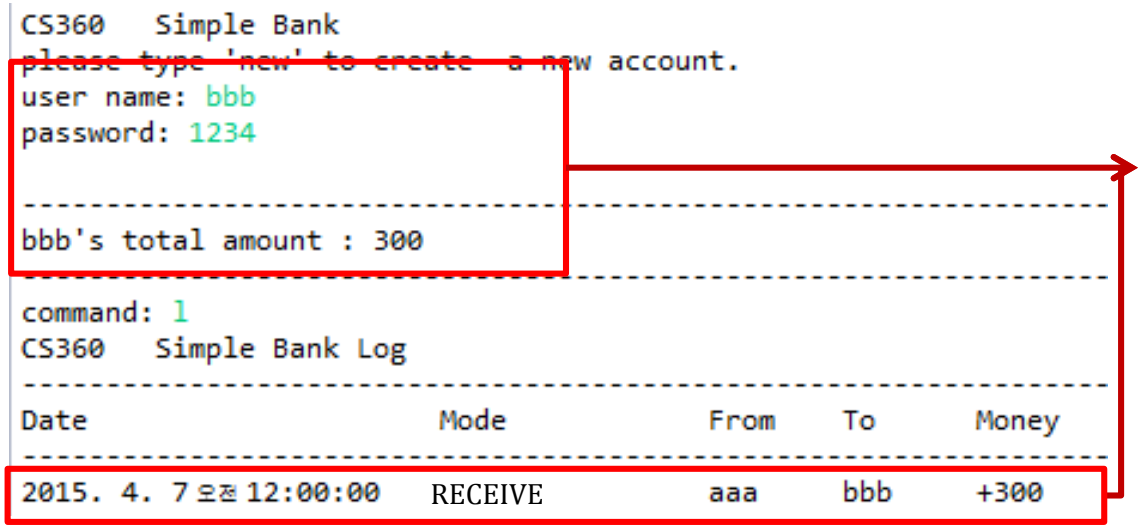
```
command: s
Enter a user for sending : bbb
Send Moeny : 300

-----------------------------------------------------------------
aaa's total amount : 200
-----------------------------------------------------------------
command: l
CS360   Simple Bank Log
-----------------------------------------------------------------
Date                    Mode          From    To      Money
-----------------------------------------------------------------
2015. 4. 7 오전 12:00:00    DEPOSIT       null    aaa     +1000
2015. 4. 7 오전 12:00:00    WITHDRAW      aaa     null    -500
2015. 4. 7 오전 12:00:00    DEPOSIT       null    aaa     +600
2015. 4. 7 오전 12:00:00    DEPOSIT       null    aaa     +200
2015. 4. 7 오전 12:00:00    WITHDRAW      aaa     null    -800
2015. 4. 7 오전 12:00:00    SEND          aaa     bbb     -300
```

type 's'; input user id; input money;
show the result &
record this result in LOGS

You should handle an inconsistent money input

Homework #4 (cont'd)

- HW4-5) Command: send(s) & transaction control (cont'd)



Also, user "bbb" can see the result & the received record

Submission

- **Files to submit**
  - 1. JAVA (*.java) and C# (*.cs) files that implement the Simple Bank
    - » i.e., you have to implement **two versions** (JAVA using JDBC and C# using OLE DB/ADO .NET) of Simple Bank
  - 2. Archive them into [student ID].zip and upload it to course homepage (KLMS)

- **Evaluation**
  - You will get points if your source codes are complied successfully
  - You will get points if your program find the right answers and is written correctly
  - Do not cheat others. Both of them will get no point

Submission (cont'd)

- **Due date**
  - April 15 (Wed), 12 pm.
  - Delay is not accepted
- **TA info.**
  - **Kwang-Hee Lee** (email : kwanghee@dbserver.kaist.ac.kr)
- **Assignment #4 TA Office hour**
  - **Tue, 2:30~4:00(pm) in N1, 403**
- **Please use KLMS Q/A board and check FAQ board**
  - **if you have a question.**

References

- Related files(example.java, example.cs) are uploaded in KLMS
- JAVA Installation
  - (Korean version) http://blog.naver.com/5suhyeon/220299496827
  - (English version) http://docs.oracle.com/javase/8/docs/technotes/guides/install/windows_jdk_install.html#CHDEBCCJ
- JDBC
  - JAVA Platform, Standard Edition 8 API Specification : http://docs.oracle.com/javase/8/docs/
  - documentation : http://docs.oracle.com/javase/8/docs/technotes/guides/jdbc/index.html
  - java.sql.Date & java.util.Date CLASS : https://stackoverflow.com/questions/16206285/comparing-date-in-java-string-form-with-jdbc-date-type
- OLE DB/ADO.NET
  - Visual C# : http://msdn2.microsoft.com/ko-kr/library/kx37x362.aspx
  - OLE DB : http://msdn.microsoft.com/en-us/library/6d9ew87b(v=vs.71).aspx